Accelerating Parameter Inference in Diffusion-Reaction Models of Glioblastoma Using Physics-Informed Neural Networks

Andy Zhu*

Project advisors: Jonathan Vo† and Dr. John Lowengrub‡

Abstract

Glioblastoma is an aggressive brain tumor with cells that infiltrate and proliferate rapidly into surrounding brain tissue. Current mathematical models of glioblastoma growth capture this behavior using partial differential equations (PDEs) that are simulated via numerical solvers—a highly-efficient implementation can take about 80 seconds to complete a single forward evaluation. However, clinical applications of tumor modeling are often framed as inverse problems that require sophisticated numerical methods and, if implemented naively, can lead to prohibitively long runtimes that render them inadequate for clinical settings. Recently, physics-informed neural networks (PINNs) have emerged as a novel method in scientific machine learning for solving nonlinear PDEs. Compared to traditional solvers, PINNs leverage unsupervised deep learning methods to minimize residuals across mesh-free domains, enabling greater flexibility while avoiding the need for complex grid constructions. Here, we describe and implement a general method for solving time-dependent diffusion-reaction PDE models of glioblastoma and inferring biophysical parameters from numerical data via PINNs. We evaluate the PINNs over patient-specific geometries, accounting for individual variations with diffusion mobilities derived from pre-operative MRI scans. Using synthetic data, we demonstrate the performance of our algorithm in patient-specific geometries. We show that PINNs are capable of solving parameter inference inverse problems in approximately one hour, expediting previous approaches by 20–40 times owing to the robust interpolation capabilities of machine learning algorithms. We anticipate this method may be sufficiently accurate and efficient for clinical usage, potentially rendering personalized treatments more accessible in standard-of-care medical protocols.

1 Introduction

Glioblastoma multiforme (GBM) is the most aggressive and common malignant primary intracranial tumor, comprising 57% of gliomas and 48% of malignant central nervous system tumors [1]. GBM is characterized

---
*Northwood High, Irvine, CA 92620 (`azhu529@gmail.com`)

†Center for Complex Biological Systems, University of California, Irvine, CA 92697 (`vojh@uci.edu`)

‡Departments of Mathematics and Biomedical Engineering, Center for Cancer Systems Biology, Center for Complex Biological Systems, University of California, Irvine, CA 92697 (`jlowengr@uci.edu`)

by infiltration and microvascular proliferation beyond visible lesions on medical scans, with a median survival time of 15 months [2]. While there is no cure, treatments include resection, radiotherapy, and chemotherapy; these procedures are generally derived from population studies and fail to account for individual patient details (for instance, current radiotherapy plans construct a simple uniform margin beyond visible tumor volumes, neglecting the anisotropic and patient-specific nature of tumor growth).

Mathematical models that incorporate raw patient data may address this shortcoming with computational analyses, offering doctors a better understanding of a tumor's prognosis, morphology, and response to various treatments [3,4]. These models can be used to rationally generate personalized RT plans, surgical operations, and chemotherapy dosages [5–8] that reduce overtreatment, increase the targeted accuracy of malignant tissue, and improve patient outcomes.

Existing models often formulate tumor growth with partial differential equations (PDEs); these equations are used to construct inverse problems in which biophysical parameters of the model are inferred from empirical tumor observations [9–16]. Solving the patient-specific inverse problem thereby calibrates the model, enabling personalized prognoses and treatment decisions. However, simulation methods remain a challenge—even highly-efficient numerical solvers can require 60–180 seconds [3, 17] to produce a single forward evaluation. With approximate [10, 18] and fully Bayesian [3] calibrations requiring tens of thousands of evaluations that translate to weeks of computation, this cost is prohibitive at the scale needed for widespread clinical viability.

Over the last two decades, deep learning has driven revolutionary advancements in image classification, drug discovery, and reinforcement learning [19]. Only recently, however, has attention been diverted toward solving PDEs with these methods. Introduced in [20], physics-informed neural networks (PINNs) map input parameters to the solution of a differential equation, constructing a loss function defined from the PDE's residual. In contrast to data-driven methods [21], PINNs do not serve as surrogates for externally-generated data, but instead directly learn the PDE by expressing differential operators with automatic differentiation (AD) [22]. Here, we consider the application of PINNs to glioblastoma growth models.

First, we develop a PINN to solve a forward problem simulating tumor growth over a designated time period, mapping spatial locations $\boldsymbol{x}$ at times t to output tumor concentrations. Second, following [23], we modify the forward problem by changing the loss function to solve a parameter inference problem in which unknown biophysical values of the model are recovered by the network using the same training algorithm in the forward problem. Third, we demonstrate the performance of the approach using synthetic data generated by the solution of the forward problem. We show that PINNs can resolve the parameters to 15% accuracy within 1.25 hours for an approximately 20–40-fold speedup compared to traditional methods, lowering the cost and increasing the potential accessibility of personalized cancer treatments in clinical settings.

2 Tumor Simulator

2.1 MRI Dataset

We consider de-identified pre-operative patient MRI scans acquired from the McConnell Brain Imaging Centre and the Montreal Neurological Institute at McGill University [3]. Included in the dataset are white matter, gray matter, and cerebrospinal fluid segmentations, which we use to construct a patient-specific domain. Fig. 1 presents a single slice of the data.
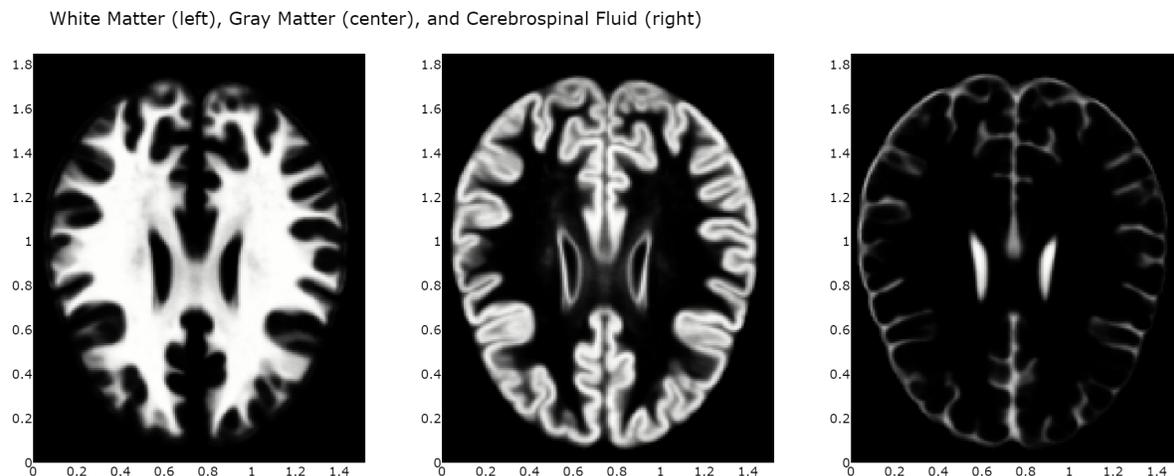
White Matter (left), Gray Matter (center), and Cerebrospinal Fluid (right)

Figure 1: Axial view of white matter (left), gray matter (center), and cerebrospinal fluid (right). Scans were acquired at a resolution of $1~mm^3$ with dimension $193 \times 193 \times 229$. Length scale: 1 unit = 10 cm.

2.2 Mathematical Growth Model

We consider the Fisher-Kolmogorov diffusion-reaction equation with logistic growth [6, 24, 25], selecting this model for its ability to describe infiltration and proliferation of tumor cells, and because it serves as a baseline for more complicated approaches.

Let $u \in [0, 1]$ denote normalized tumor cell density, $D(\boldsymbol{x})$ denote the diffusion coefficient representing the infiltration of the tumor cells at position $\boldsymbol{x}$, and ρ denote the proliferation rate of the tumor cells. Let $\boldsymbol{\Omega}$ denote the brain anatomy as defined by the MRI scans. Then we construct the following differential equation:

$$\frac{\partial u}{\partial t} = \overbrace{\nabla \cdot [D(\boldsymbol{x})\nabla u]}^{\text{Diffusion}} + \overbrace{\rho u(1 - u)}^{\text{Proliferation}} \text{ in } \boldsymbol{\Omega}, \tag{1}$$

where D, ρ serve as downstream parameters describing more complex behavior (e.g. individual cell movements [26]). Equation (1) mathematically describes a "traveling wave" solution [27] in which the tumor expands radially outwards by the diffusion term while increasing in density to a carrying capacity by the proliferation term. Fisher's equation [28] approximates the speed of the traveling wave's "edge" as $2\sqrt{D\rho}$, which we use to verify our simulations.

Table 1 summarizes the salient elements of the mathematical model.

Symbol	Value	Units	Definition
Variables			
$\boldsymbol{x}$		mm	Spatial coordinate
t		days	Temporal coordinate
u		-	Tumor concentration
Parameters			
D_w	0.13	mm^2 days^{-1}	Measure of random tumor dispersal in white matter
p_w	$-$	-	Proportion of white matter
D_g	0.013	mm^2 days^{-1}	Measure of random tumor dispersal in gray matter
p_g	$-$	-	Proportion of gray matter
ρ	0.025	days^{-1}	Measure of net tumor proliferation from natural processes
$\boldsymbol{x_{ic}}$	$(105, 140, 99)$	mm	Tumor origin location
T	$\{150, 300\}$	days	Simulation duration

Table 1: Variables and parameters in the diffusion-reaction growth model.

2.2.1 Diffusion and Proliferation Constants

The diffusion coefficient is patient-specific, defined as a weighted sum of gray and white matter [6]: $D = p_w D_w + p_g D_g$ where the values of p_w and p_g denote heterogeneous tissue proportions that are determined from the patient MRI scans [3]. This accounts for observations that tumor growth is more pronounced along white matter tracts. We assume $D_w = 10 D_g$ [29] and apply a bilinear interpolation [30] to approximate the white and gray matter concentrations at fine resolutions, finding this method to be within our computational restraints and less prone to oscillations than higher-order alternatives.

We set $D_w = 1.3 \times 10^{-1}$ mm^2 days^{-1} and $\rho = 2.5 \times 10^{-2}$ days^{-1} [25], consistent with a standard high-grade glioma. We consider simulation durations of 150 and 300 days, finding the former is sufficient in the patient-specific case to form a tumor with a radius of roughly 1.5 cm. Using Fisher's approximation, we estimate the radial velocity of the tumor's expansion to be 0.11 mm days^{-1}, implying radii of 1.7 and 3.4 cm for the tumors at 150 and 300 days, respectively.

2.2.2 Initial Condition

The tumor concentration is initialized as a Gaussian distribution about a point $\boldsymbol{x_{ic}}$ [24]:

$$\text{IC}(\boldsymbol{x}) = 0.1 \exp(-10\|\boldsymbol{x} - \boldsymbol{x_{ic}}\|^2). \tag{2}$$

The distribution is chosen such that the initial tumor has an approximate radius of 5 mm.

2.2.3 Boundary Condition

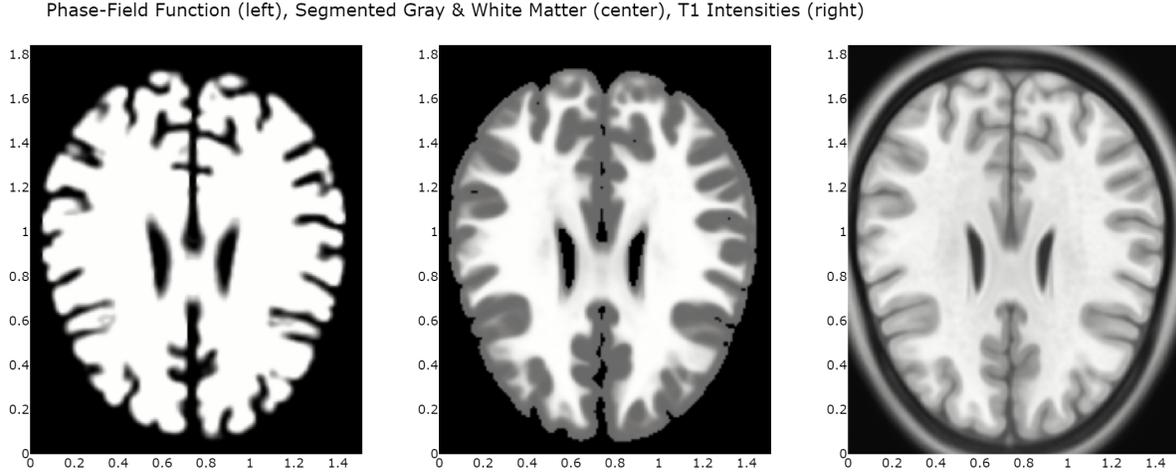Phase-Field Function (left), Segmented Gray & White Matter (center), T1 Intensities (right)

Figure 2: Axial plane view of phase-field function (left), diffusion coefficients (center), and T1 scans (right). The phase-field function defines the boundary of the simulation domain. Length scale: 1 unit = 10 cm.

Since tumor cells cannot penetrate the skull or enter the ventricles [6, 31] we impose a no-flux boundary condition:

$$\nabla u \cdot \vec{n} = 0 \text{ on } \partial\mathbf{\Omega}, \tag{3}$$

where $\vec{n}$ represents the normal surface vector and $\partial\mathbf{\Omega}$ denotes the boundary of the skull and ventricles. We enforce these boundary conditions using the diffuse domain method (DDM) [32–34], allowing us to express the complex brain geometry. To generate the phase-field function ϕ from a binary segmentation of combined gray and white matter, we construct a simple rectangular prism containing the cropped brain geometry $\mathbf{\Omega}_R := [0, 138] \times [0, 167] \times [0, 123][\text{mm}^3]$ and solve the following Cahn-Hilliard equation [3]:

$$\frac{\partial\phi}{\partial t} = \nabla \cdot [\phi(1 - \phi) \cdot \nabla(g'(\phi) - \epsilon^2\Delta\phi)], \tag{4}$$

where

$$\phi(t = 0) = \begin{cases} 1 & \text{in } \mathbf{\Omega}, \\ 0 & \text{in } \mathbf{\Omega}_R \setminus \mathbf{\Omega}. \end{cases} \tag{5}$$

We take $g(\phi) = \frac{1}{4}\phi^2(1 - \phi)^2$ as a double-well potential and $\epsilon = 0.1$. The equation is simulated in time until a sufficiently thin interface forms between the brain geometry and surrounding space (Fig. 2, left). Following the DDM, Eq. (1) is reformulated as [32]:

$$\frac{\partial}{\partial t}(\phi u) = \nabla \cdot (D\phi\nabla u) + \phi\rho u(1 - u). \tag{6}$$

This modification to the PDE implicitly enforces the boundary condition in Eq. (3).

2.3 Physics-Informed Neural Network

Next, we use a neural network to learn the PDE subject to patient-specific domains and boundary conditions. In particular, we construct a PINN with spatial and temporal inputs. Spatial values are selected by sampling a radially-symmetric uniform distribution centered at $\boldsymbol{x_{ic}}$ while temporal values are uniformly randomly sampled from the interval $[0, T]$[days]. We define the input space as θ_p with the neural network operation $u(\theta_p)$ mapping to tumor concentrations. The neural network is implemented with DeepXDE [23], a deep learning library built from TensorFlow [30]. We also apply a hard-constraint [23] to the PINN output $\hat{u}$ such that $u(\boldsymbol{x}, t = 0; \boldsymbol{\theta}) = \mathrm{IC}(\boldsymbol{x})$:

$$u(\boldsymbol{x}, t; \boldsymbol{\theta}) := \hat{u}(\boldsymbol{x}, t; \boldsymbol{\theta})t + \mathrm{IC}(\boldsymbol{x}). \tag{7}$$
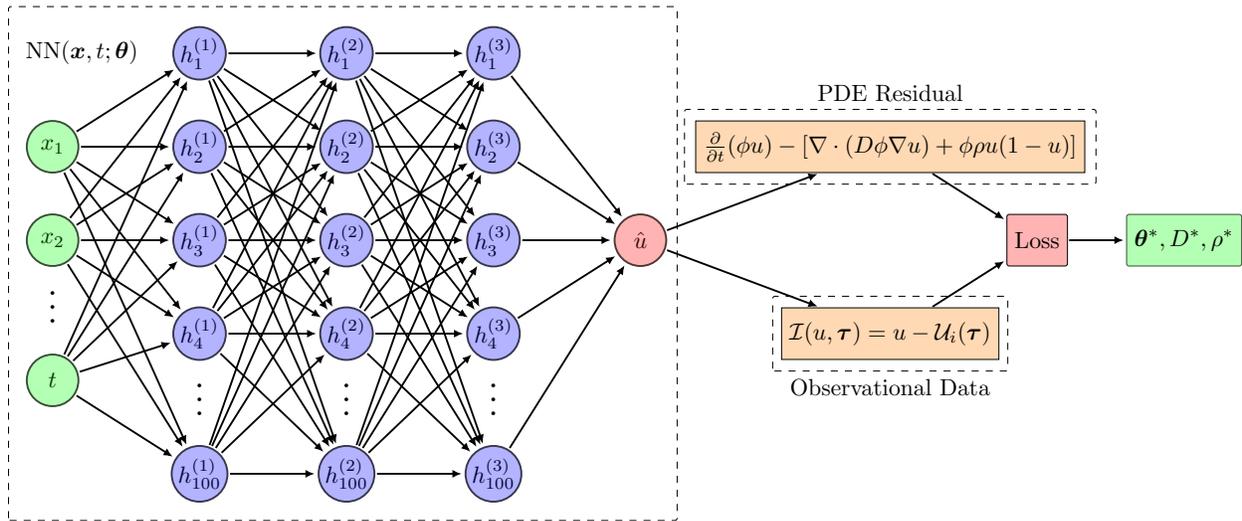
Figure 3: Diagram of a physics-informed neural network with three hidden layers used for parameter inference. The differential equation and observational data are used to concurrently optimize $\boldsymbol{\theta}$, D, and ρ, yielding $\boldsymbol{\theta}^*$, D^*, and ρ^*, respectively.

Let $\mathcal{T}_f \subset \theta_p$ denote a set of training data and $\boldsymbol{\theta}$ denote the parameters of the neural network. We construct the loss function from the mean-squared residuals of the differential equation:

$$\mathcal{L}_f(\boldsymbol{\theta}; \mathcal{T}_f) = \frac{1}{|\mathcal{T}_f|} \sum_{\mathcal{T}_f} \|\frac{\partial}{\partial t}(\phi u) - [\nabla \cdot (D\phi\nabla u) + \phi\rho u(1-u)]\|_2^2. \tag{8}$$

Thus minimizing $\mathcal{L}_f(\boldsymbol{\theta}; \mathcal{T})$ approximates the differential equation. We train the network on this loss function with two gradient-based optimizers: the Adam [35] and Broyden-Fletcher-Goldfarb-Shanno (BFGS) [36] algorithms. Automatic differentiation is used to express each term of the PDE (e.g. $\frac{\partial}{\partial t}(\phi u)$, $\phi\rho u(1-u)$).

We select a feedforward neural network with three hidden layers of 100 neurons each and input/output layers corresponding to θ_p and $u(\theta_p)$, respectively. We use the hyperbolic tangent for the activation, set the learning rate to 1.0×10^{-3}, and initialize $\boldsymbol{\theta}$ from a Glorot distribution [37]. In the patient-specific forward problem, we batch train 100,000 epochs with the Adam scheme followed by the BFGS scheme. Simulations were performed on NVIDIA Tesla V100s.

2.4 Parameter Inference

PINNs can be readily modified to solve inverse problems for parameter inference [23]: given a dataset $\mathcal{T}_i \in \theta_p$ with corresponding observations $\mathcal{U}_i$, we define a loss metric:

$$\mathcal{L}_i(\boldsymbol{\theta}, D, \rho; \mathcal{T}_i) = \frac{1}{|\mathcal{T}_i|} \sum_{\boldsymbol{\tau} \in \mathcal{T}_i} ||\mathcal{I}(u, \boldsymbol{\tau})||_2^2 \tag{9}$$

where $\mathcal{I}(u, \boldsymbol{x})$ is the elementwise difference between the observational data $\mathcal{U}_i$ and the predicted neural network inferences. Note that D and ρ have been embedded into the loss function; we aim to determine these values from empirical data. We initialize $D_w = 1.0 \times 10^{-1}$ mm^2 days^{-1} and $\rho = 1.0 \times 10^{-2}$ days^{-1} [38] as guesses that are successively inferred with the differential equation and additional information provided from $\mathcal{T}_i$. Hence the parameter inference loss function is

$$\mathcal{L}(\boldsymbol{\theta}, D, \rho; \mathcal{T}) = w_f \mathcal{L}_f(\boldsymbol{\theta}, D, \rho; \mathcal{T}_f) + w_i \mathcal{L}_i(\boldsymbol{\theta}, D, \rho; \mathcal{T}_i), \tag{10}$$

where w_f, w_i are weights. Here, we choose $w_f = w_i = 1$, finding it suitable for obtaining convergence. Because this algorithm is iterative, we establish a stopping criterion by tracking the relative changes every 1,000 epochs for the biophysical constants across a 10,000 epoch window. We stop if for epoch N,

$$\sum_{k=0}^{9} \frac{|D_{N+1000(k+1)} - D_{N+1000k}|}{|D_{N+1000k}|} + \sum_{k=0}^{9} \frac{|\rho_{N+1000(k+1)} - \rho_{N+1000k}|}{|\rho_{N+1000k}|} < \epsilon_{tol}, \tag{11}$$

where ϵ_{tol} is a user-supplied tolerance. We train the PINN over 800,000 epochs, halving the learning rate to 5.0×10^{-4} after 400,000 epochs to decelerate the network's fluctuations. A schematic of the PINN for parameter inference is presented in Fig. 3.

3 Results

We consider two forward problems in 2D: one on a circular domain with a radius of 5 cm and another on a patient-specific domain constructed from MRI scans (Fig. 2, axial plane). Using the synthetic data generated from solutions to the forward problems, we infer the parameter values of D and ρ.

3.1 Simple Domain

3.1.1 Forward Problem

We set $T = 300$ days and solve the forward problem in Eq. (6) over the radially-symmetric domain via a finite-difference numerical solver [39]. We let $\phi = 0.5 + 0.5 \tanh(50 - r)$ where r is the distance from a point to the origin, creating a boundary at $r = 50$ mm. In Fig. 4, we present the solutions u at 30-day intervals; we observe that the tumor expands radially outward, grows to a maximum concentration of about $u = 0.9$, and has a radius of approximately 3.5 cm.
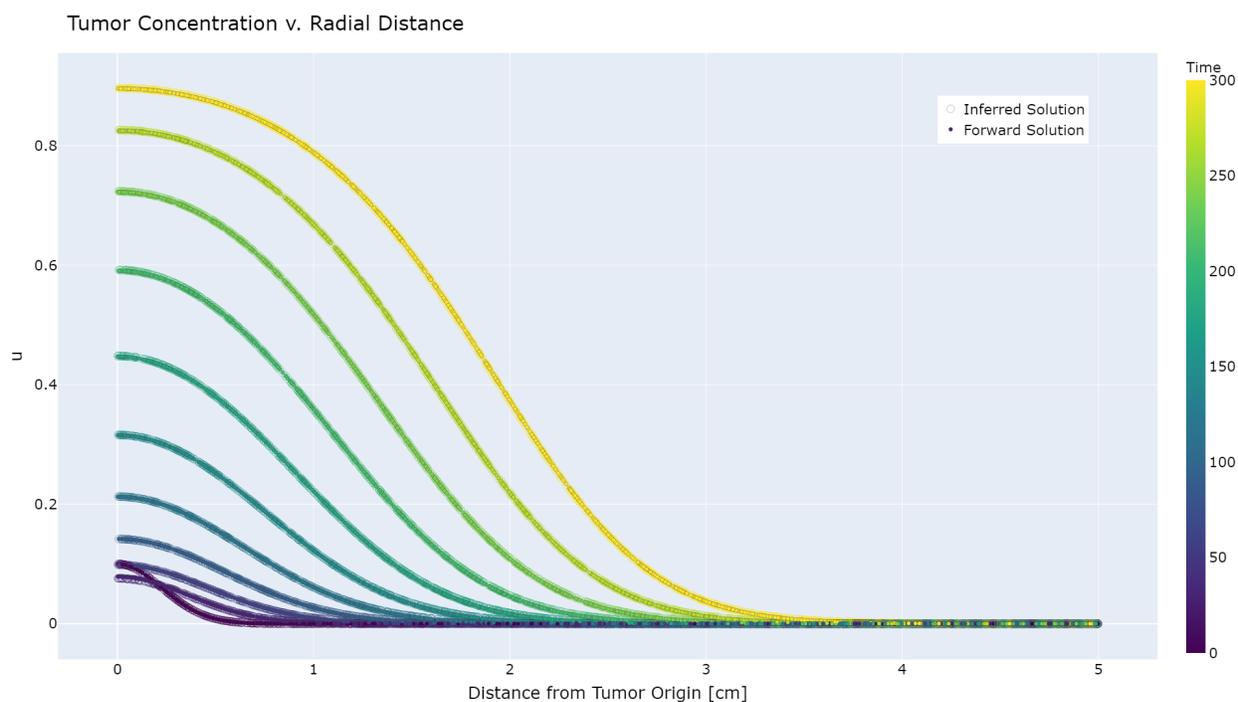
Figure 4: Plot of u vs. radial distance colored by time to Day 300. Solid: radially-symmetric, finite difference numerical solver. Circles: PINN solutions with inferred parameters. Solutions presented in 30-day intervals.

3.1.2 Parameter Inference

From the forward problem solution in Sec. 3.1.1, we sample 20,000 new inputs and observations up to $T = 300$ days. We infer parameters from this data for 800,000 epochs and consider tolerances from 0.04 to 0.20. The total runtime is 250 minutes.
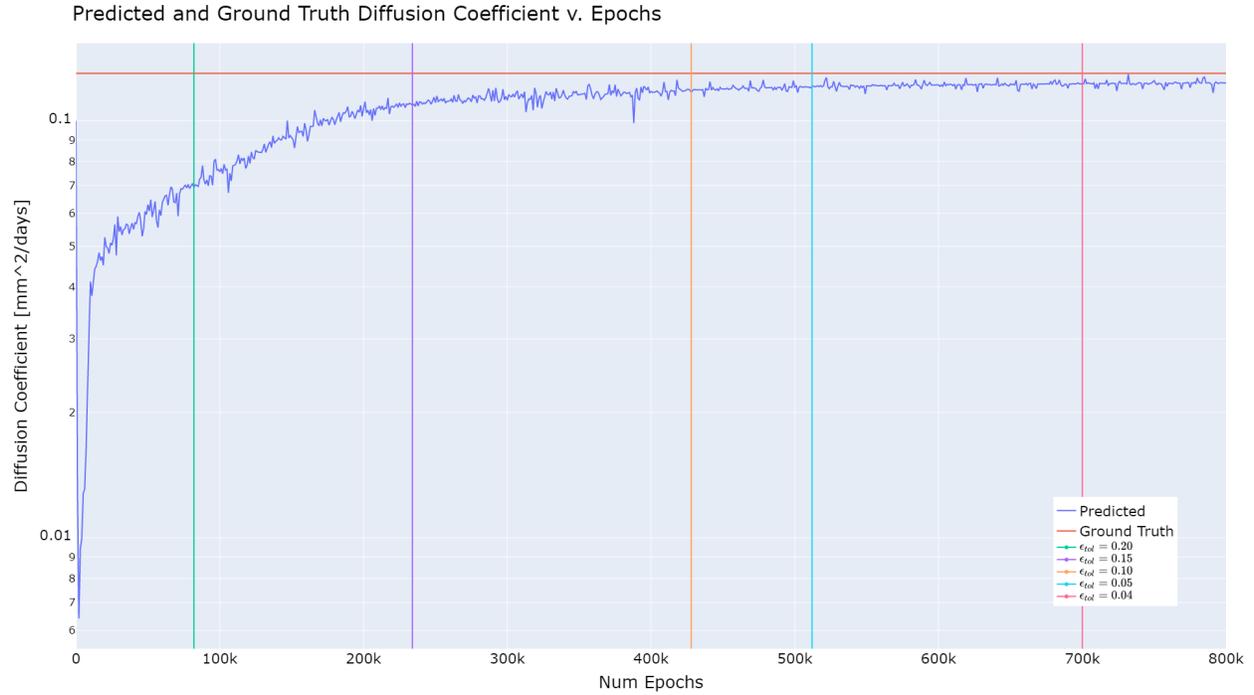
Predicted and Ground Truth Diffusion Coefficient v. Epochs

Figure 5: Predicted and actual diffusion coefficients against epochs. Initial value of $D = 1.0 \times 10^{-1}$ mm^2 $days^{-1}$ is optimized to $D \approx 1.23 \times 10^{-1}$ mm^2 $days^{-1}$. Tolerances and corresponding epochs shown as colored vertical lines.

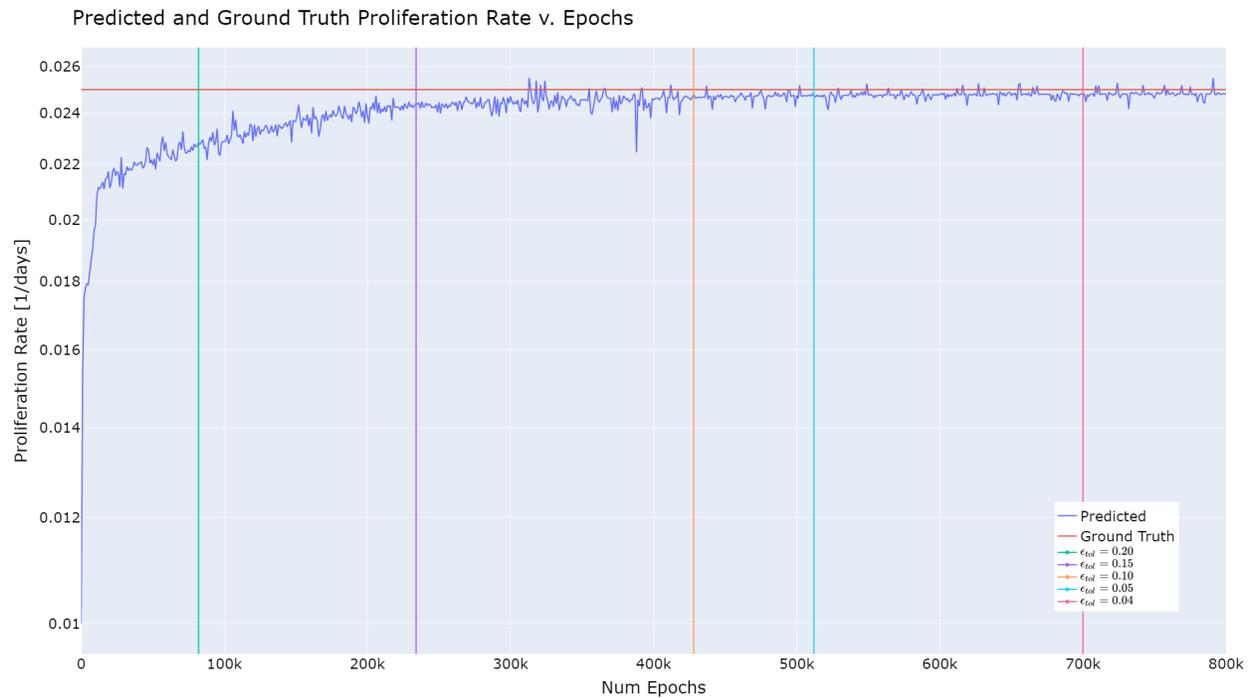Predicted and Ground Truth Proliferation Rate v. Epochs

Figure 6: Predicted and actual proliferation rates against epochs. Initial value of $\rho = 1.0 \times 10^{-2}$ $days^{-1}$ is optimized to $\rho \approx 2.48 \times 10^{-2}$ $days^{-1}$. Tolerances and corresponding epochs shown as colored vertical lines.

The PINN solution to Eq. (6) with inferred parameters and the ground truth radially-symmetric solution are shown in Fig. 4. Notably, the agreement is excellent; the PINN solution (solid) and the ground truth (circle) exhibit a high degree of overlap. The mean-squared difference between the trained PINN's solution and the radially-symmetric finite difference solution in Sec. 3.1.1 is $\mathcal{L}_i = 2.71 \times 10^{-7}$ while the final PDE residual is $\mathcal{L}_f = 2.19 \times 10^{-6}$.

From Figs. 5 and 6, we observe that the final predicted values of D and ρ differ from their true values by 5 and 1 percent, respectively. Further inspection of the graphs reveals rapid initial movements in both parameters for the first few thousand epochs, followed by gradual asymptotic increases toward the ground truth values. From Fig. 7, both parameters have relative errors within approximately 15% after 234,000 epochs (1.25 hours of training). The inferred parameters continue to fluctuate before demonstrating more stable behavior after 400,000 epochs. Table 2 presents the runtimes at various tolerance levels, along with corresponding epochs and relative errors in D and ρ.

Tolerance (ϵ_{tol})	Epoch	Time (mins.)	D_w % Error	ρ % Error
0.20	82000	26	46.69	8.74
0.15	234000	75	16.00	2.36
0.10	428000	136	9.97	0.74
0.05	512000	163	6.78	1.06
0.04	700000	223	5.54	0.81

Table 2: Epochs, times, and relative errors for diffusion and proliferation constants against error tolerances in the radially-symmetric domain.
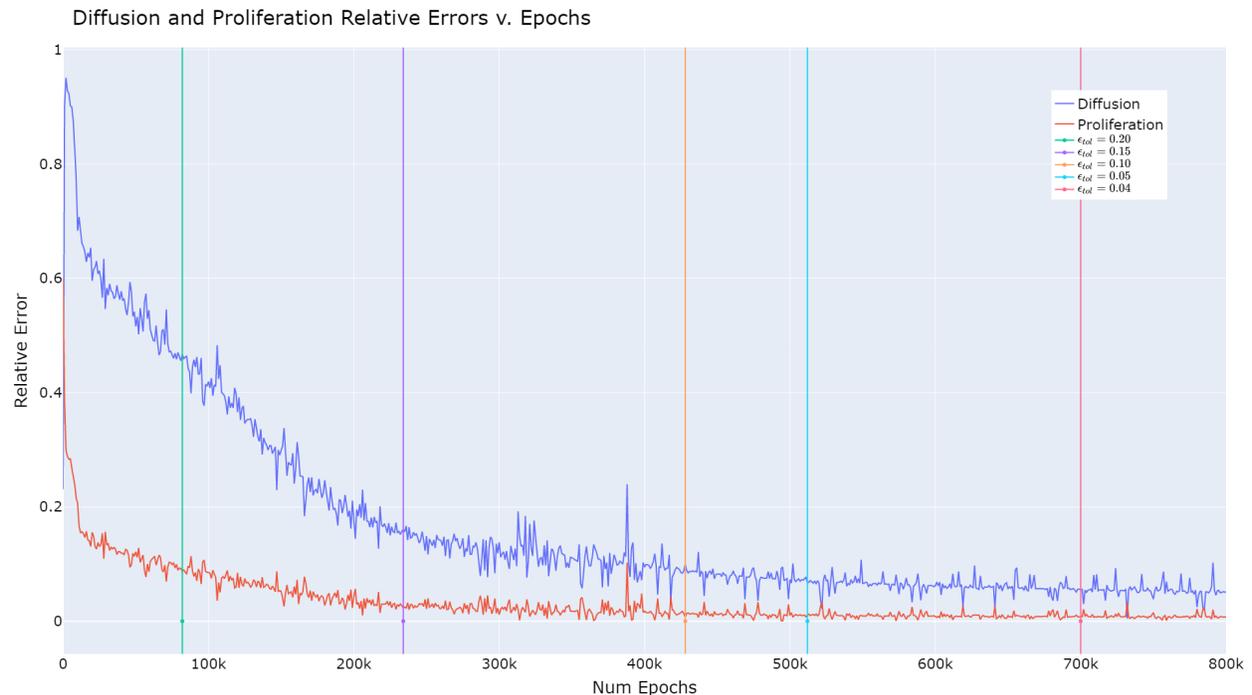

Figure 7: Relative errors between predicted and ground truth diffusion coefficients and proliferation rates versus epochs in radially-symmetric domain.
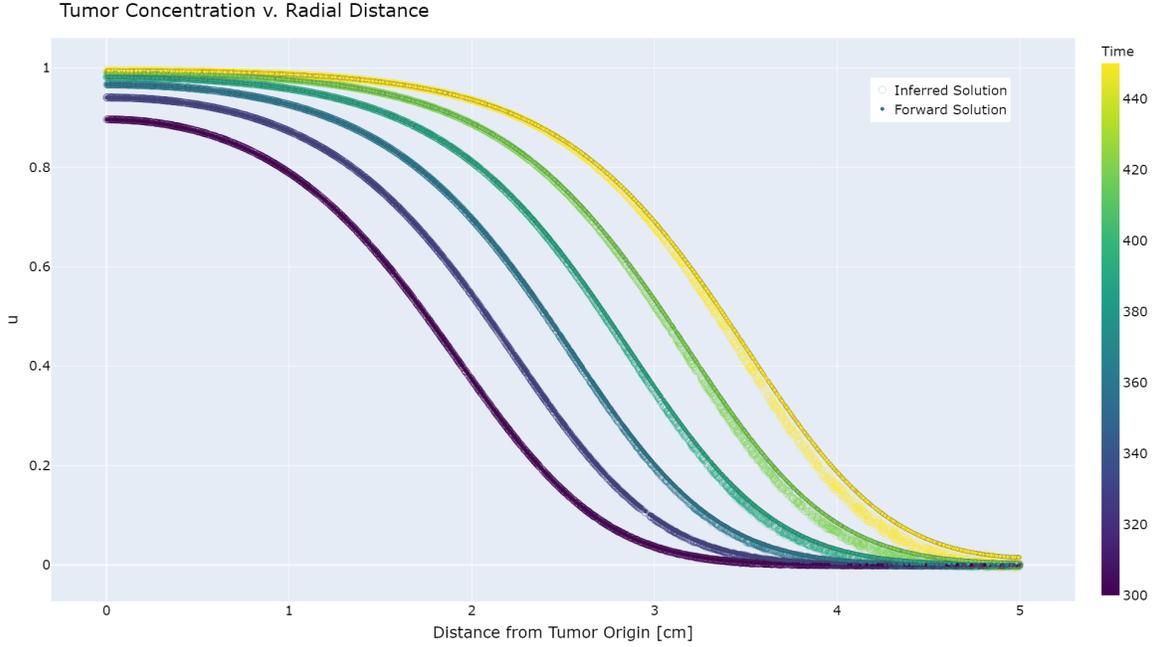
Figure 8: Plot of u vs. radial distance colored by time from Day 300 to 450. Solid: radially-symmetric, finite difference numerical solver. Circles: trained parameter inference PINN solutions on new time domain. Solutions presented in 30-day intervals.
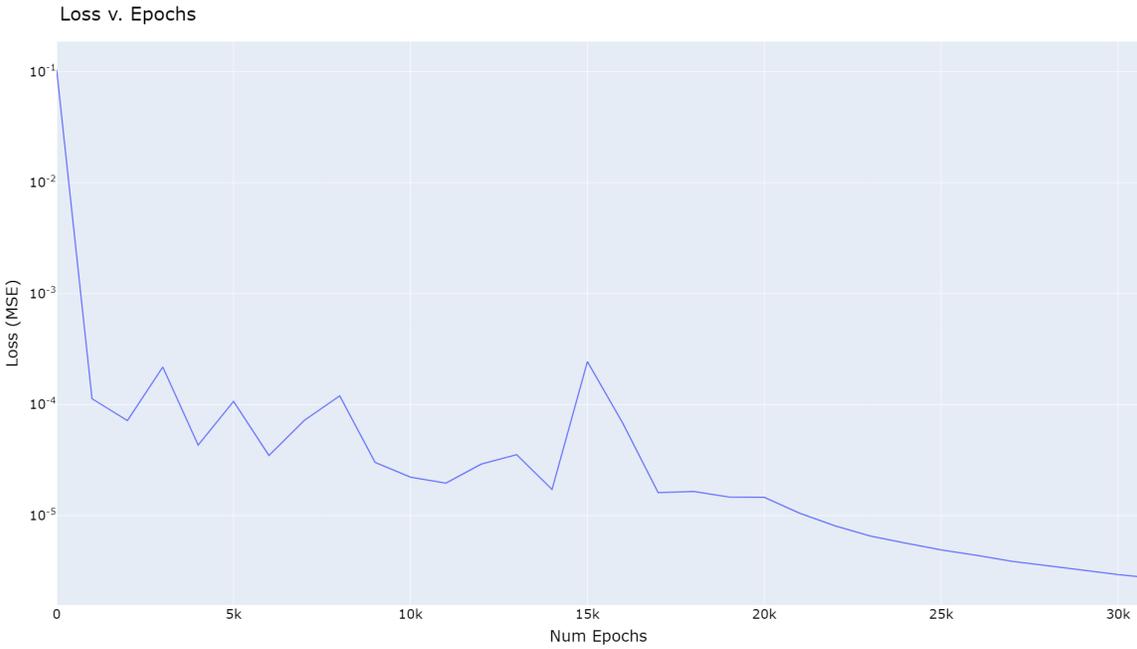

Figure 9: Plot of mean-squared loss versus training epoch while minimizing residual with fixed inferred parameter. The network trains for 20,000 epochs with the Adam scheme followed by L-BFGS-B, finishing in 8 minutes with a final loss of 2.81×10^{-6}.

Additionally, we test how well the PINN with inferred parameters can predict the solution at later times. We solve Eq. (6) from $T = 300$ to $T = 450$ using the PINN with the inferred parameters held fixed. After 30,000 epochs, we report a total runtime of 8 minutes and a final loss of 2.81×10^{-6}. Thus, the re-trained

PINN is able to approximate unseen data with high accuracy. The comparison between the numerical solver and re-trained PINN is presented in Fig. 8. The mean-squared difference between the trained PINN and numerical solution is 6.62×10^{-5}.

3.2 Patient-Specific Domain

3.2.1 Forward Problem

We solve Eq. (6) via PINNs in a patient-specific geometry defined from a cross-section of MRI scans (Fig. 2) and simulate the tumor evolution for $T = 150$ days. We sample 20,000 collocation points for the training data and present the solution in Fig. 10 along with training logs in Fig. 12.
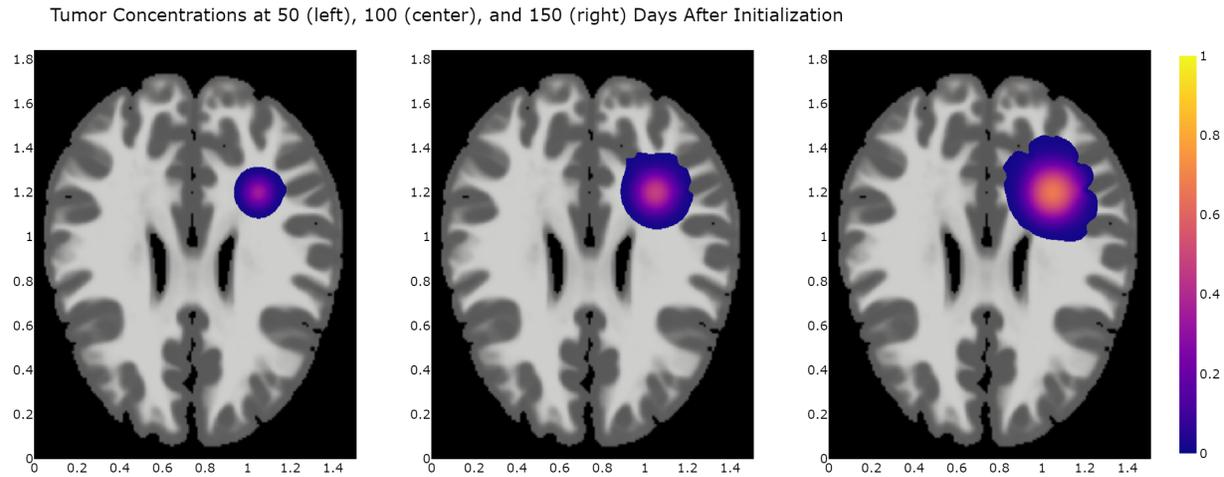

Figure 10: Axial view of tumor concentrations at 50-day intervals in patient-specific geometry as generated by a forward PINN solver. Tumor growth is anisotropic. Length scale: 1 unit = 10 cm.
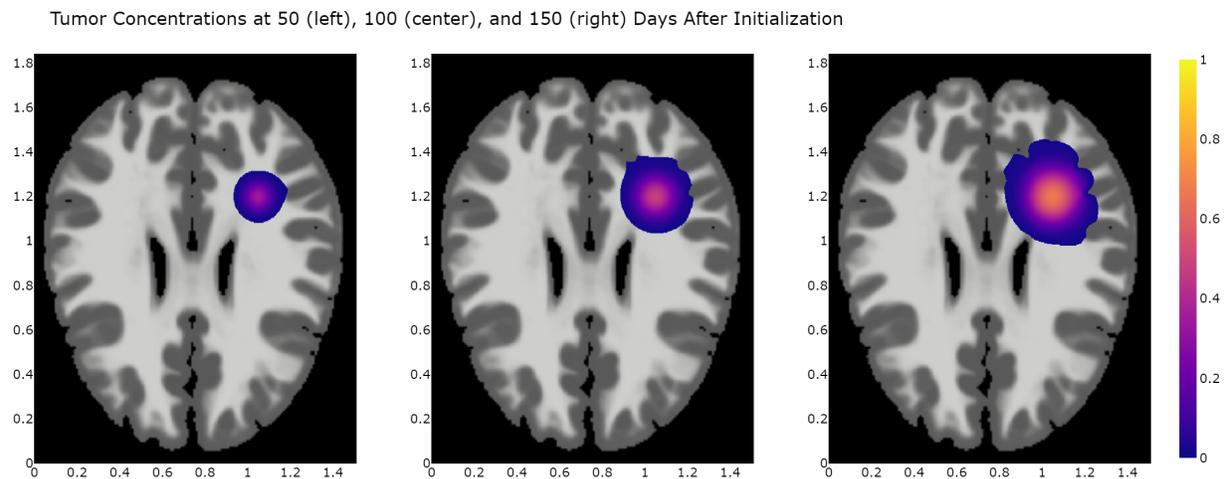

Figure 11: Axial view of tumor concentrations at 50-day intervals in patient-specific geometry as learned by the trained PINN with inferred parameters. The concentrations exhibit close agreement with ground truth concentrations in Fig. 10. Length scale: 1 unit = 10 cm.
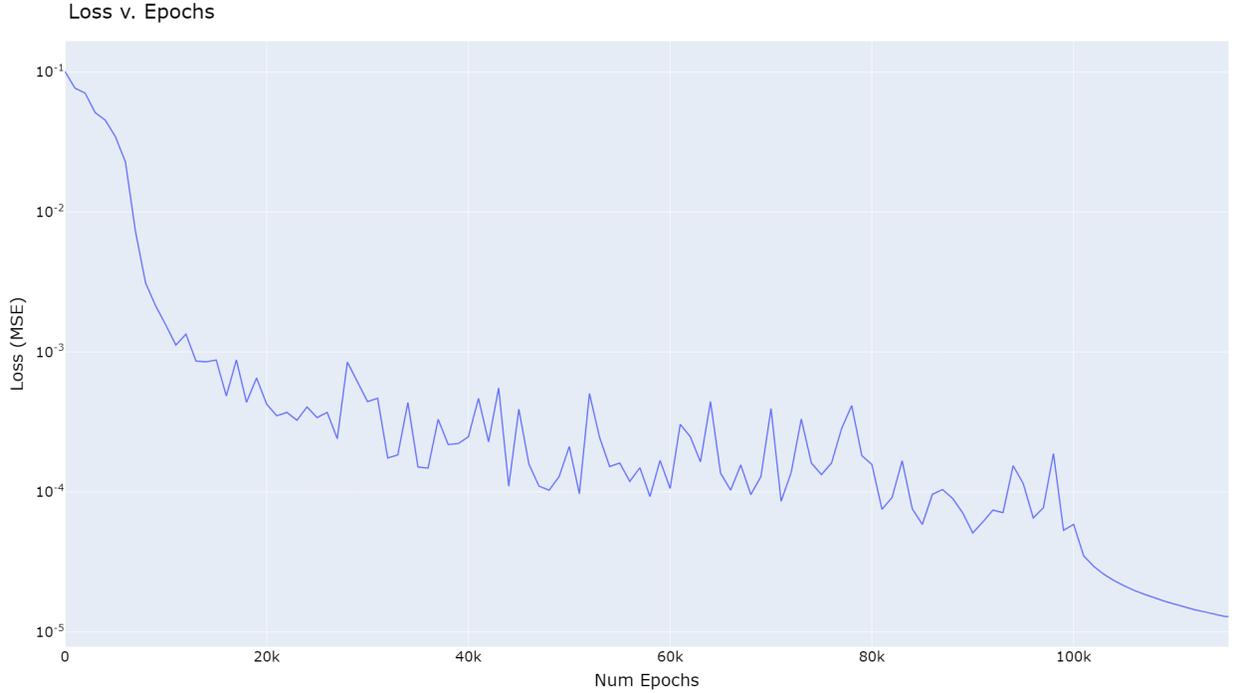
Figure 12: Plot of mean-squared loss versus training epoch. The network trains for 100,000 epochs with the Adam scheme followed by the BFGS scheme, finishing in 50 minutes with a final loss of 1.29×10^{-5}.

Figure 10 reveals anisotropic diffusion, with infiltration more rapid along white matter tracts, as the tumor expands across the right frontal lobe. The tumor radius we estimated at Day 150 is 1.5 cm. The tumor avoids infiltrating the skull and ventricles, observing the no-flux condition. The network rapidly optimizes the PDE residual for the first 20,000 epochs followed by slower progress until the BFGS scheme converges the loss to the order of 10^{-5} (Fig. 12). Table 3 presents the losses, diffusion coefficients, and proliferation rates for the last 5,000 epochs of parameter inference.

Epoch	Loss	D_w [mm^2 days^{-1}]	ρ [days^{-1}]
795000	$[4.19 \times 10^{-6}, 4.96 \times 10^{-7}]$	1.26×10^{-1}	2.48×10^{-2}
796000	$[2.05 \times 10^{-5}, 1.58 \times 10^{-6}]$	1.28×10^{-1}	2.46×10^{-2}
797000	$[1.48 \times 10^{-5}, 1.33 \times 10^{-6}]$	1.25×10^{-1}	2.49×10^{-2}
798000	$[2.79 \times 10^{-5}, 1.57 \times 10^{-6}]$	1.24×10^{-1}	2.50×10^{-2}
799000	$[2.37 \times 10^{-5}, 8.33 \times 10^{-7}]$	1.26×10^{-1}	2.47×10^{-2}
800000	$[3.78 \times 10^{-6}, 5.36 \times 10^{-7}]$	1.26×10^{-1}	2.48×10^{-2}

Table 3: Losses, diffusion coefficients, and proliferation rates for the last 5,000 epochs of parameter inference in the patient-specific domain. The loss is formatted as $[\mathcal{L}_f(\boldsymbol{\theta}; \mathcal{T}_f), \mathcal{L}_i(\boldsymbol{\theta}, D, \rho; \mathcal{T}_i)]$, presenting the PDE and observational residuals.

3.2.2 Parameter Inference

From the forward problem in Sec. 3.2.1, we sample 26,830 new inputs and observations up to $T = 150$ days. We infer parameters from this data for 800,000 epochs and consider tolerances from 0.07 to 0.20. The total runtime is 450 minutes.
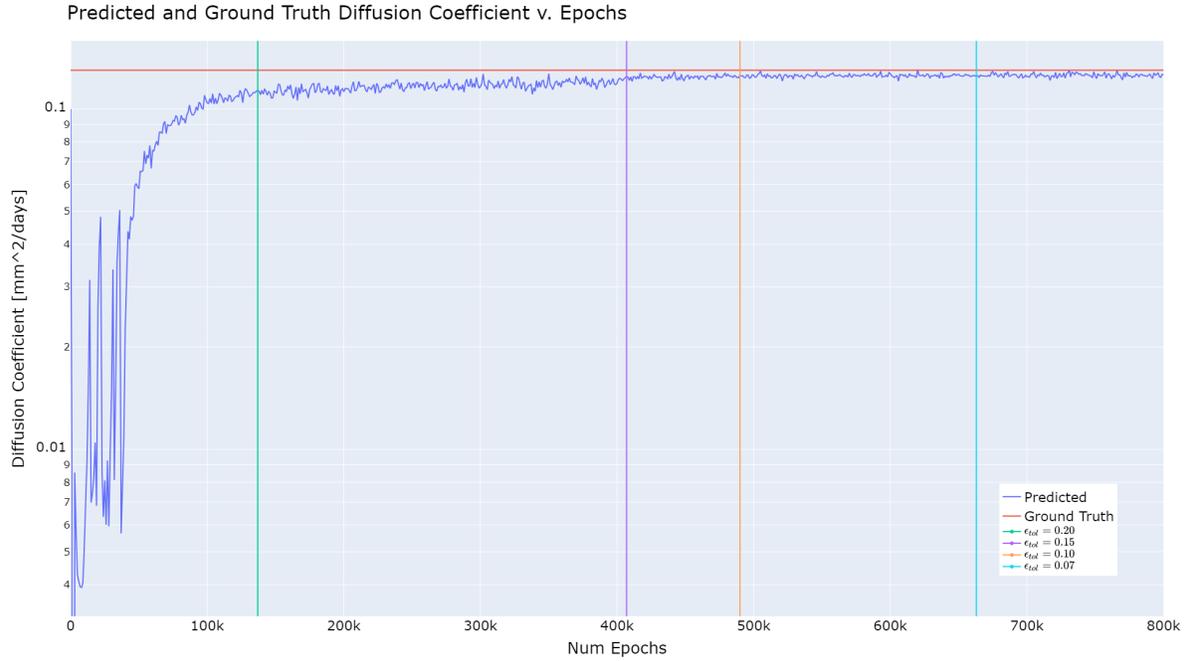
Figure 13: Predicted and actual diffusion coefficients against epochs in patient-specific geometry. Initial value of $D_w = 1.0 \times 10^{-1}$ mm^2 $days^{-1}$ is optimized to $D_w \approx 1.26 \times 10^{-1}$ mm^2 $days^{-1}$. Tolerances and corresponding epochs shown as colored vertical lines.
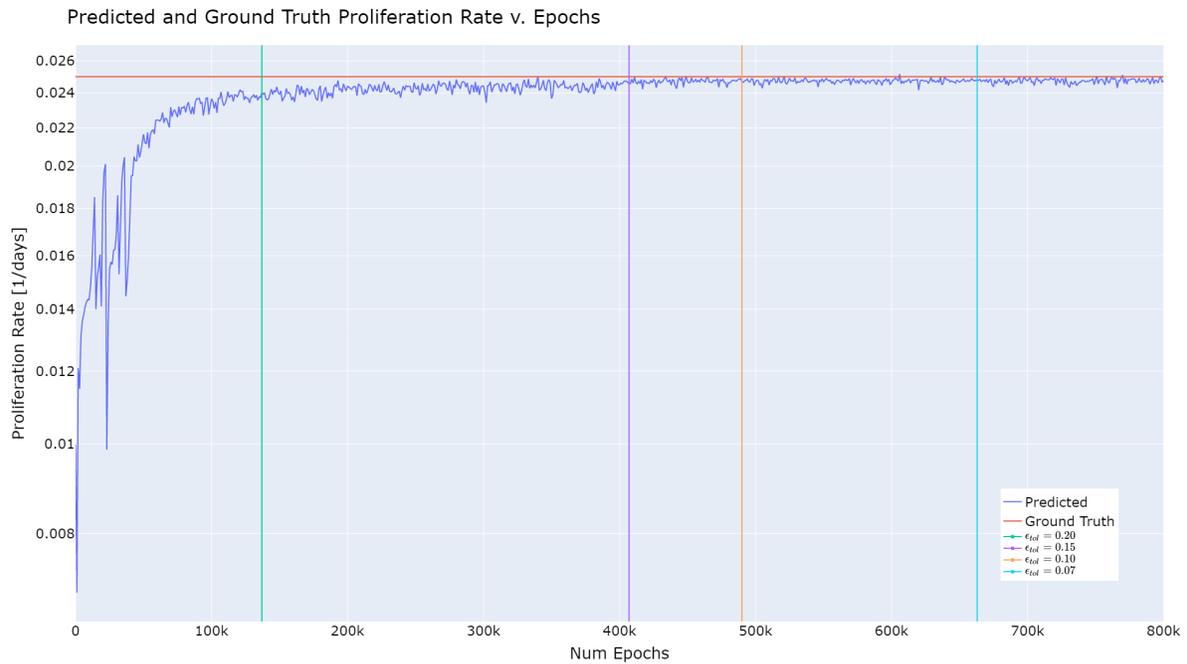

Figure 14: Predicted and actual proliferation rates against epochs in patient-specific geometry. Initial value of $\rho = 1.0 \times 10^{-2}$ $days^{-1}$ is optimized to $\rho \approx 2.48 \times 10^{-2}$ $days^{-1}$. Tolerances and corresponding epochs shown as colored vertical lines.

Figure 11 presents the PINN solutions after training and inferring the parameters from the forward solution in Sec. 3.2.1. There is excellent agreement between the PINN and ground truth tumor concentrations— the mean-squared difference between the trained PINN's solution and the forward solution in Sec. 3.2.1 is $\mathcal{L}_i = 5.36 \times 10^{-7}$ while the final PDE residual is $\mathcal{L}_f = 3.78 \times 10^{-6}$

From Figs. 13 and 14, we observe that the final predicted values of D and ρ differ from their true values by 3 and 1 percent, respectively. Further inspection of the graphs reveals rapid initial movements in both parameters for the first few thousand epochs, followed by gradual asymptotic increases to the ground truth values. From Fig. 15, both parameters have relative errors within 15% after 137,000 epochs (1.25 hours of training). Table 3 shows that the final loss is small for both the PDE and observational residuals, having reached the orders of 10^{-6} and 10^{-7}, respectively. Table 4 presents the runtime at various tolerance levels, along with corresponding epochs and relative errors in D and ρ. Strikingly, the computation times are very similar to those in the simple geometry (Table 2). The errors in D and ρ are even smaller in the patient-specific geometry, likely because the ground truth tumor concentrations were generated by the forward solver of the PINN.

Tolerance (ϵ_{tol})	Epoch	Time (mins.)	D_w % Error	ρ % Error
0.20	137000	76	12.90	3.87
0.15	407000	224	4.15	1.74
0.10	490000	271	3.54	1.28
0.07	663000	365	3.53	0.92

Table 4: Epochs, times, and relative errors for diffusion and proliferation constants against error tolerances in the patient-specific domain.
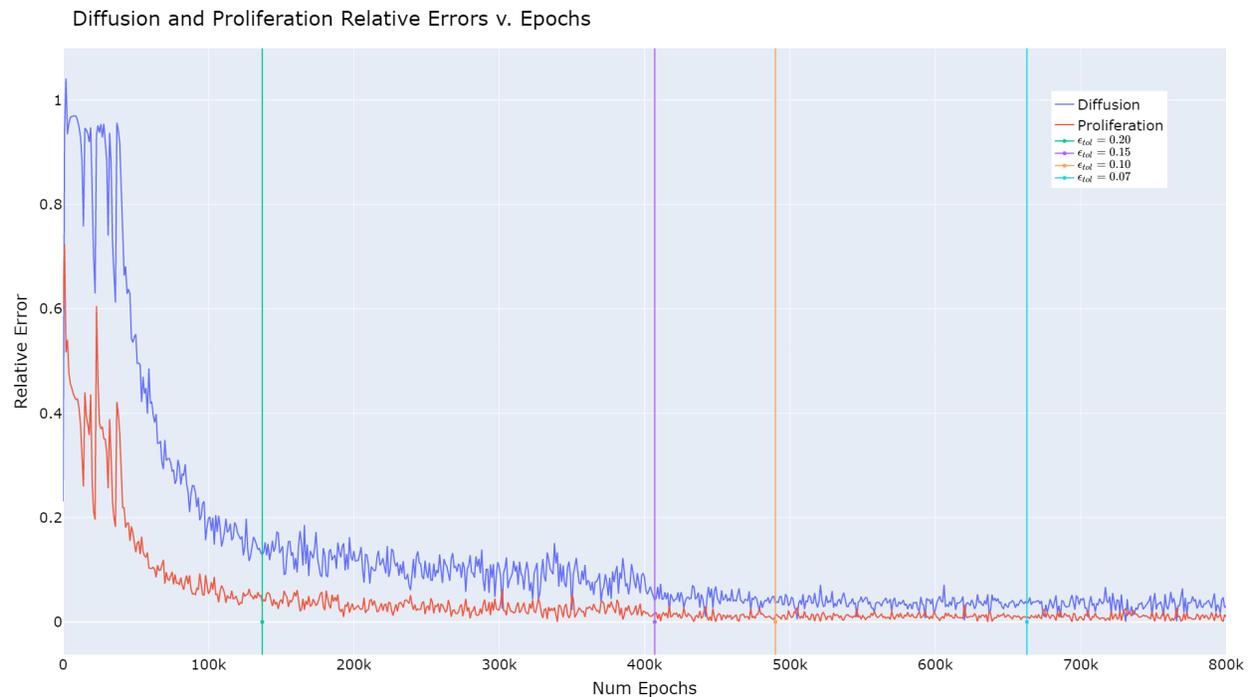

Figure 15: Relative errors between predicted and ground truth diffusion coefficients and proliferation rates versus epochs in patient-specific domain.

4 Discussion

We find that PINNs perform robustly in forward and inverse diffusion-reaction simulations of glioblastoma in both simple and patient-specific geometries. The PDE and observational losses are minimized to the orders of 10^{-6} and 10^{-7}, respectively. Close agreement between predicted and actual tumor radii (1.7 and 1.5 cm in the simple geometry, 3.4 and 3.5 cm in the patient-specific geometry) further suggests accurate learning of the differential equation. Moreover, the PINN captures the anisotropic growth of brain tumors in patient-specific geometries, implying functionality in heterogeneous domains.

In Sec. 3.1.1, we use a radially-symmetric finite difference method to provide ground truth data to train the PINN and infer the diffusion coefficient D and proliferation rate ρ. Once the network has completed training, we hold constant the predicted parameters for D and ρ and then solve a forward problem in a time domain beyond the original problem, introducing unseen data to the network. The resulting solution is highly accurate relative to the ground truth radial solution, yielding a mean-squared error on the scale of 10^{-5}, and converges in only 8 minutes. We find that continuing to train from the PINN trained over earlier spatiotemporal data yields significantly faster convergence than re-initializing a new network.

In Sec. 3.2.1, we solve forward problems via a PINN in 50 minutes; utilizing the flexibility of the mesh-free neural network, we successfully account for heterogeneous diffusion. Compared to the 60–180 seconds required by modern finite difference methods [3, 17], we agree with the author of [40] that PINNs fare poorly against traditional solvers in evaluating forward problems. Nevertheless, we solve them to construct synthetic data for the parameter inference inverse problem and ensure the functionality of PINNs over patient-specific domains.

We conclude that PINNs are highly capable of recovering biophysical parameters. Notably, the diffusion and proliferation constants are determined to within five and one percent accuracy, respectively, in both simple and patient-specific geometries with about an hour of computation. In future work, we plan to generate data from a different numerical solver in time for the patient-specific geometry, thus creating an independent source to construct the inference dataset. We may also consider adding varying degrees of noise to gauge the robustness of the PINN approach to parameter inference.

We find that the error tolerances based on relative changes of iterates provide an effective condition for halting the training process; relying on relative changes over a moving window of 10,000 epochs grants the network greater resistance to minor fluctuations. From Tables 2 and 4, we observe that lower tolerances tend to imply better agreements between the predicted and ground truth values, though at the cost of increased training time. We also find that reducing the learning rate is crucial to improving convergence: after halving the learning rate at 400,000 epochs, we observe significantly reduced variation in the predicted values on Figs. 6, 7, 11, and 12. We caution that the selection of the tolerance is largely heuristic, and can be improved provided one is willing to continue training the network; here, we prioritize speed for medical applicability. In future works, we may consider using dropout [41] and B-PINNs (Bayesian PINNs) [42] to construct uncertainty bounds for the problem parameters, providing an alternative to user-supplied tolerances.

We observe that while traditional numerical methods in [3, 17] can efficiently solve a single forward problem, they must be used to solve thousands of problems in order to address inverse problems—particularly using

Bayesian approaches—translating to lengthy computational costs. In contrast, PINNs bypass the need for forward solvers by concurrently minimizing the residual from the PDE and the differences between predicted and observed data. Here, we recover patient-specific problem parameters to within 15% accuracy in as little as 1.25 hours (Secs. 3.1.2, 3.2.2); in [3], the authors suggest 6,000 forward simulations of their numerical solver are sufficient for inference, equating to 100–300 hours of computation. Thus we achieve anywhere from an 80–240-fold increase in speed (we note that the relative errors in their simulations for diffusion and proliferation constants were 0.2 and 20 percent, respectively). However, the inverse problems we solved involved fewer parameters than the ones in [3,11,21], including statistical variables and initial positions, and were evaluated over two dimensions, not three. Assuming the complexity of the 3D problem is $\mathcal{O}(n^3)$ and the 2D problem is $\mathcal{O}(n^2)$, we conservatively estimate their numerical solver would require 15–30 seconds per simulation of a 2D problem (exponentiating to 2/3). This limits our speedup to 20–40 times; nevertheless, we find that PINNs are a promising alternative to sequential forward computations in parameter inference tasks. More conservative conditions, such as training for 400,000 epochs so that relative errors are bounded within 5%, would reduce our speedup to 5–10 times. In follow-up works, we plan to incorporate these additional parameters in the inference problem and consider similar models for other solid tumors such as melanomas, colon cancer, breast cancer, and lung cancer.

It should be noted that the required runtime for parameter inference varies greatly. In [14–16], a fine-resolution reaction-diffusion system in three dimensions requires merely six hours to recover diffusion and proliferation constants within relative errors of 22 and 15 percent, respectively. The results presented here ought to be viewed as a baseline for expediting current numerical algorithms using non-traditional methods.

Extending our methodology to three dimensions is limited by the phase-field function: the discrete resolution of the MRI scans causes accidental artifacts after solving Eq. (5) (e.g. tissue connections between brain hemispheres, disrupting the boundary). Though we manually check for this in our two-dimensional simulations, a more rigorous algorithm may be needed to efficiently define the boundary in three-dimensional volumes. Moreover, effective interpolation algorithms should be explored for smoothing p_w and p_g. The linear interpolation we apply here creates discontinuities in the diffusion term, as first-order spatial derivatives of linear interpolations produce piecewise values. While we do not find that this significantly impacted our solutions, it remains desirable that a superior method be developed.

We observe that the hyperparameters of the PINNs can be improved: our chosen feedforward networks may benefit from greater depth, breadth, and training data in addressing the loss function. However, increasing any of these quantities will incur additional computational expenses, forming a trade-off between model complexity and accuracy. Other architectures such as residual or convolutional networks may be considered, along with an assortment of parameters (e.g. learning rates, batch sizes, activation functions) that can be heuristically determined. Moreover, our choice of a diffusion-reaction model can be replaced by a Fokker-Planck equation [43] or other models that may describe the tumor's progression by accounting for more biophysical processes. We can also directly encode treatments such as resection, radiotherapy, and chemotherapy into the solver by changing the model equation [38, 44]. To fully validate the capabilities of this computational framework, it could be tested on a cohort of patients to investigate whether the predicted parameters and subsequently-generated treatments accurately combat *in vivo* tumor growth.

5   Conclusion

We have demonstrated that PINNs are capable of solving patient-specific diffusion-reaction equations for glioblastoma in forward and parameter inference problems, the latter holding clinical significance for its applications to personalized medicine. In our experiments, we find that PINNs require about an hour to accurately calibrate a set of biophysical parameters against empirical observations, improving upon the weeks of computational expenses incurred by certain standard approaches by 20–40 times. While this metric is not representative of all current numerical methods, we find it is a promising direction for future research. Moreover, we have extended the functionality of PINNs to arbitrary geometries, encoding patient-specific variations with the diffuse domain method. These contributions may expedite progress in computational medicine, thereby increasing the efficacy of existing treatments for glioblastoma patients. Due to the flexible nature of neural networks, this approach is readily modifiable to address other models for diseases such as breast cancer and lung cancer, offering doctors and patients alike a mathematically-guided tool to inform medical decisions.

6   Acknowledgements

References

[1] Quinn T Ostrom, Gino Cioffi, Haley Gittleman, Nirav Patil, Kristin Waite, Carol Kruchko, and Jill S Barnholtz-Sloan. CBTRUS statistical report: Primary brain and other central nervous system tumors diagnosed in the United States in 2012–2016. *Neuro-Oncology*, 21(Supplement_5):v1–v100, 2019.

[2] R Stupp, M Brada, MJ Van Den Bent, J-C Tonn, and GESMO Pentheroudakis. High-grade glioma: ESMO clinical practice guidelines for diagnosis, treatment and follow-up. *Annals of Oncology*, 25:iii93–iii101, 2014.

[3] Jana Lipková, Panagiotis Angelikopoulos, Stephen Wu, Esther Alberts, Benedikt Wiestler, Christian Diehl, Christine Preibisch, Thomas Pyka, Stephanie E. Combs, Panagiotis Hadjidoukas, Koen Van Leemput, Petros Koumoutsakos, John Lowengrub, and Bjoern Menze. Personalized radiotherapy design for glioblastoma: Integrating mathematical tumor models, multimodal scans, and Bayesian inference. *IEEE Transactions on Medical Imaging*, 38(8):1875–1884, 2019.

[4] Cecilia Suarez, Felipe Maglietti, Mario Colonna, Karina Breitburd, and Guillermo Marshall. Mathematical modeling of human glioma growth based on brain topological structures: study of two clinical cases. *PloS One*, 7(6):e39616, 2012.

[5] David A Hormuth, Karine A Al Feghali, Andrew M Elliott, Thomas E Yankeelov, and Caroline Chung. Image-based personalization of computational models for predicting response of high-grade glioma to chemoradiation. *Scientific Reports*, 11(1):1–14, 2021.

[6] Kristin R Swanson, Ellsworth C Alvord Jr, and JD Murray. A quantitative model for differential motility of gliomas in grey and white matter. *Cell Proliferation*, 33(5):317–329, 2000.

[7] Jan Unkelbach, Bjoern H Menze, Ender Konukoglu, Florian Dittmann, Matthieu Le, Nicholas Ayache, and Helen A Shih. Radiotherapy planning for glioblastoma based on a tumor growth model: improving target volume delineation. *Physics in Medicine & Biology*, 59(3):747, 2014.

[8] Russell C Rockne, Andrew D Trister, Joshua Jacobs, Andrea J Hawkins-Daarud, Maxwell L Neal, Kristi Hendrickson, Maciej M Mrugala, Jason K Rockhill, Paul Kinahan, Kenneth A Krohn, et al. A patient-specific computational model of hypoxia-modulated radiation resistance in glioblastoma using 18f-fmiso-pet. *Journal of the Royal Society Interface*, 12(103):20141174, 2015.

[9] James D Murray. Mathematical biology: I. an introduction. interdisciplinary applied mathematics. *Mathematical Biology, Springer*, 17, 2002.

[10] Ivan Ezhov, Jana Lipkova, Suprosanna Shit, Florian Kofler, Nore Collomb, Benjamin Lemasson, Emmanuel Barbier, and Bjoern Menze. Neural parameters estimation for brain tumor growth modeling. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 787–795. Springer, 2019.

[11] Ivan Ezhov, Tudor Mot, Suprosanna Shit, Jana Lipkova, Johannes C Paetzold, Florian Kofler, Chantal Pellegrini, Marcel Kollovieh, Fernando Navarro, Hongwei Li, et al. Geometry-aware neural solver for fast

bayesian calibration of brain tumor models. *IEEE Transactions on Medical Imaging*, 41(5):1269–1278, 2021.

[12] K Sakthivel, S Gnanavel, N Barani Balan, and K Balachandran. Inverse problem for the reaction diffusion system by optimization method. *Applied mathematical modelling*, 35(1):571–579, 2011.

[13] Olivier Clatz, Maxime Sermesant, P-Y Bondiau, Hervé Delingette, Simon K Warfield, Grégoire Malandain, and Nicholas Ayache. Realistic simulation of the 3-d growth of brain tumors in mr images coupling diffusion with biomechanical deformation. *IEEE transactions on medical imaging*, 24(10):1334–1346, 2005.

[14] Klaudius Scheufele, Shashank Subramanian, and George Biros. Fully automatic calibration of tumor-growth models using a single mpmri scan. *IEEE Transactions on Medical Imaging*, 40(1):193–204, 2020.

[15] Shashank Subramanian, Klaudius Scheufele, Naveen Himthani, and George Biros. Multiatlas calibration of biophysical brain tumor growth models with mass effect. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 551–560. Springer, 2020.

[16] Klaudius Scheufele, Shashank Subramanian, Andreas Mang, George Biros, and Miriam Mehl. Image-driven biophysical tumor growth model calibration. *SIAM Journal on Scientific Computing: a publication of the Society for Industrial and Applied Mathematics*, 42(3):B549, 2020.

[17] Matthieu Lê, Hervé Delingette, Jayashree Kalpathy-Cramer, Elizabeth R Gerstner, Tracy Batchelor, Jan Unkelbach, and Nicholas Ayache. Bayesian personalization of brain tumor growth model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 424–432. Springer, 2015.

[18] Matthieu Lê, Hervé Delingette, Jayashree Kalpathy-Cramer, Elizabeth R Gerstner, Tracy Batchelor, Jan Unkelbach, and Nicholas Ayache. Personalized radiotherapy planning based on a computational tumor growth model. *IEEE Transactions on Medical Imaging*, 36(3):815–825, 2016.

[19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[20] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[21] Ivan Ezhov, Jana Lipkova, Suprosanna Shit, Florian Kofler, Nore Collomb, Benjamin Lemasson, Emmanuel Barbier, and Bjoern Menze. Neural Parameters Estimation for Brain Tumor Growth Modeling. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11765 LNCS, 2019.

[22] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18, 2018.

[23] Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1), 2021.

[24] P Tracqui, GC Cruywagen, DE Woodward, GT Bartoo, JD Murray, and EC Alvord Jr. A mathematical model of glioma growth: the effect of chemotherapy on spatio-temporal growth. *Cell Proliferation*, 28(1):17–31, 1995.

[25] Hana LP Harpold, Ellsworth C Alvord Jr, and Kristin R Swanson. The evolution of mathematical modeling of glioma proliferation and invasion. *Journal of Neuropathology & Experimental Neurology*, 66(1):1–9, 2007.

[26] Christina H Wang, Jason K Rockhill, Maciej Mrugala, Danielle L Peacock, Albert Lai, Katy Jusenius, Joanna M Wardlaw, Timothy Cloughesy, Alexander M Spence, Russ Rockne, et al. Prognostic significance of growth kinetics in newly diagnosed glioblastomas revealed by combining serial imaging with a novel biomathematical model. *Cancer Research*, 69(23):9133–9140, 2009.

[27] Kristin Rae Swanson. *Mathematical modeling of the growth and control of tumors.* University of Washington, 1999.

[28] Nanako Shigesada and Kohkichi Kawasaki. *Biological invasions: theory and practice.* Oxford University Press, UK, 1997.

[29] Bjoern H Menze, Koen Van Leemput, Antti Honkela, Ender Konukoglu, Marc-André Weber, Nicholas Ayache, and Polina Golland. A generative approach for image-based modeling of tumor growth. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 735–747. Springer, 2011.

[30] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[31] James D Murray. *Mathematical biology II: spatial models and biomedical applications*, volume 3. Springer New York, 2001.

[32] Julien Kockelkoren, Herbert Levine, and Wouter-Jan Rappel. Computational approach for modeling intra-and extracellular dynamics. *Physical Review E*, 68(3):037702, 2003.

[33] Karl Yngve Lervåg and John Lowengrub. Analysis of the diffuse-domain method for solving PDEs in complex geometries. *arXiv preprint arXiv:1407.7480*, 2014.

[34] X Li, John Lowengrub, A Rätz, and A Voigt. Solving PDEs in complex geometries: a diffuse domain approach. *Communications in Mathematical Sciences*, 7(1):81, 2009.

[35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[36] Roger Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2013.

[37] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[38] Anyue Yin, Dirk Jan AR Moes, Johan GC van Hasselt, Jesse J Swen, and Henk-Jan Guchelaar. A review of mathematical models for tumor dynamics and treatment resistance evolution of solid tumors. *CPT: Pharmacometrics & Systems Pharmacology*, 8(10):720–737, 2019.

[39] David Zwicker. py-pde: A python package for solving partial differential equations. *Journal of Open Source Software*, 5(48):2158, 2020.

[40] Stefano Markidis. The old and the new: Can physics-informed deep-learning replace traditional linear solvers? *Frontiers in Big Data*, page 92, 2021.

[41] Pierre Baldi and Peter J Sadowski. Understanding dropout. *Advances in Neural Information Processing Systems*, 26:2814–2822, 2013.

[42] Liu Yang, Xuhui Meng, and George Em Karniadakis. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 425:109913, 2021.

[43] Amanda Swan, Thomas Hillen, John C Bowman, and Albert D Murtha. A patient-specific anisotropic diffusion model for brain tumour spread. *Bulletin of Mathematical Biology*, 80(5):1259–1291, 2018.

[44] JD Murray. Glioblastoma brain tumours: estimating the time from brain tumour initiation and resolution of a patient survival anomaly after similar treatment protocols. *Journal of Biological Dynamics*, 6(sup2):118–127, 2012.